# COMP 170: Introduction to Object-Oriented Programming (Section 001)
# Spring 2008 Course Information & Syllabus

**Instructor:** R. I. Greenberg
Computer Science Department
Loyola University
Water TowerCampus, Lewis Towers 512E
820 North Michigan Avenue
Chicago, Illinois 60611-2147

**Phone:** (312)915-7981 for messages; also (773)508-3782 on Monday/Wednesday/Friday
**Email:** rig@cs.luc.edu **Home page:** http://www.cs.luc.edu/~rig
**TA tutoring schedule:** See `http://www.cs.luc.edu/academics/services/tutoring`.

**Lectures:** Monday, Wednesday, and Friday 10:25–11:15 am (lab Wednesday 12:35–1:25) in DH-339 (also for lab).
Sometimes lecture notes or a summary may be available on the web. Other than that, if you have to miss a class, get notes from another student; mine are typically pieced together from more than one place with a lot of metacomments, which makes it hard for anybody but me to follow them. Also get copies of any missed handouts (available on the web site). The handouts are numbered sequentially, starting with handout 0. On handout 0, you need to fill in some information and return it to me promptly so you can be on the email list and get access to the web site for the course.

**Office Hours:** In DH-225: Monday 2:30–3:30pm, Wednesday 2:00–3:30pm, and Friday 12:30–2:00pm.
These are the guaranteed times to find me except as announced in advance. You should also feel free to look for me at other times or make appointments.

**Course Description:** This course will serve as an introductory programming course for some students and as a course that will greatly deepen programming skills for students who have some prior programming background. A key focus of this course is on the object-oriented style of programming as facilitated by the Java programming language. This is a programming-intensive course that includes a required weekly lab component. Students should leave the course with a good understanding of such basic programming language topics as variables, data types, input/output, conditional execution, repetition, arrays, and subprograms, as well as a facility with object-oriented programming in Java that will provide a proper foundation for further programming-related courses in the computer science major.

**Prerequisites:** Prerequisite of MATH 117 OR corequisite of COMP 150 or COMP 163.

**Textbook:** David J. Barnes and Michael Kölling. *Objects First With Java.* Pearson Education, third edition, 2006.

**Course Requirements:** There will be several homework assignments, three midterm exams, and a final project. The weightings within the semester grade will be: Exam I 15%, Exam II 15%, Exam III 20%, Final Project 20%, Other Homework and Labs 30%.

**Homework:** Only homework turned in by class time on the due date is guaranteed to be graded. Any special circumstances that cause difficulty in meeting the deadlines should be brought to the attention of the instructor in advance.

**Exams:** The midterm exams, tentatively scheduled for week 5 and week 8 and week 14, are 50 minutes long. The final exam period (9:00-11:00 am on Wednesday, April 30) will be used for presentations on final projects.

**Collaboration:** *No* collaboration is permitted on exams. *Collaboration* on homework is acceptable, but *copying* is not! (Safeguard your files and printouts.) You may discuss solution techniques with other students, but you must write up your solutions independently. If you obtain a solution through research, e.g., in the library, credit your source and write up the solution in your own words.

**Tentative Course Outline and Approximate Schedule:**

Recommended readings from the text are shown on a weekly basis. (When selected sections or subsections are listed, it is assumed that you will include the introduction of the corresponding chapter or section.)

Detailed schedule TBA. Basic coverage at least Chapters 1, 2, 3, 4, 5, 7, and 8.

1. (1/14) Course and Computer Administrivia/Logistics, Working with Classes/Objects in BlueJ, Source Code, esp. fields and constructors for classes. Chapter 1, Sections 2.1–3
2. (1/23) Parameter Passing, Assignments, Methods, Printing, Conditionals, Local Variables. Sections 2.4–18.
3. (1/28) Object Interaction, Debugger. Chapter 3.
4. (2/4) Collections of Objects. Sections 4.1–9
5. (2/11) Object Collections continued. Sections 4.10–13. Exam I on Chapters 1–3.
6. (2/18) Using Libary Classes. `String` and `Random` Classes. Sections 5.1–5.
7. (2/25) Maps, Sets, Documentation, Access Modifiers, etc. Sections 5.6–14.
8. (3/10) Coupling, Cohesion, Code Duplication. Sections 7.1–6. Exam II on Chapters 4–5.
9. (3/17) Responsibility-Driven Design, Refactoring. Sections 7.7–16.
10. (3/26) Inheritance. Sections 8.1–6.
11. (3/31) Subtyping, Polymorphic Variables, Casting, etc. Sections 8.7–11
12. (4/7) Static and Dynamic Type, Overriding, Dynamic Method Lookup, Method Polymorphism. Sections 9.1–6.
13. (4/14) Protected Access, more on Inheritance and Overriding. Sections 9.8–10. Abstract Classes and Methods. Sections 10.1–4.
14. (4/21) Exam III on Chapters 7–9. Multiple Inheritance, Interfaces. Sections 10.5–8.